Open MPI and the Roadrunner Project

David Daniel, CCS-1; Ralph Castain, ISR-4; Brian Barrett, Sandia National Laboratories; Galen Shipman, Oak Ridge National Laboratory

Large-scale parallel scientific applications are, with few exceptions, based on the Message Passing Interface (MPI) [1], which is a standardized communication library allowing the coordination of, and data exchange between, the constituent processes of an application.

The Open MPI Project [2] is an open source MPI implementation that is developed and maintained by a consortium of academic, research, and industry partners. Open MPI is therefore able to combine the expertise, technologies, and resources from all across the high-performance computing community with the goal of creating the best MPI library available. Open MPI offers advantages for system and software vendors, application developers, and computer science researchers.

Open MPI and Los Alamos. A team in CCS-1 was a founding member of the Open MPI Project and led much of the development through the first several releases, leveraging experience gained in the development and deployment of LA-MPI [3] (the MPI implementation currently in use on the Pink, Flash, and Lightning Linux production systems). Consequently, more than other MPI implementations, Open MPI has been driven by the needs of the LANL computational science community, specifically in the areas of portability, consistent user interface, and performance on systems of interest to the Laboratory.

Of particular importance for the Roadrunner project is the issue of scalability, that is, the ability to run a parallel science application efficiently on thousands of processors. As an example, consider how the memory used by an MPI library scales with the number of processors. Figure 1 compares memory usage of Open MPI 1.0 with another MPI implementation (MVAPICH v0.9.5) on systems using an InfiniBand network [4]. Many MPI implementations are aimed at relatively small clusters–up to a few hundred processors–so concern about the memory used per process is a secondary consideration. The result is that when scaled to thousands of processors an ever-increasing fraction of memory is consumed by the MPI library rather than being available to an application. Not so for Open MPI, which was designed from the beginning to scale well on very large systems, and has been successfully used on Sandia National Laboratories' Thunderbird Linux cluster at scales in excess of 8,000 processes.

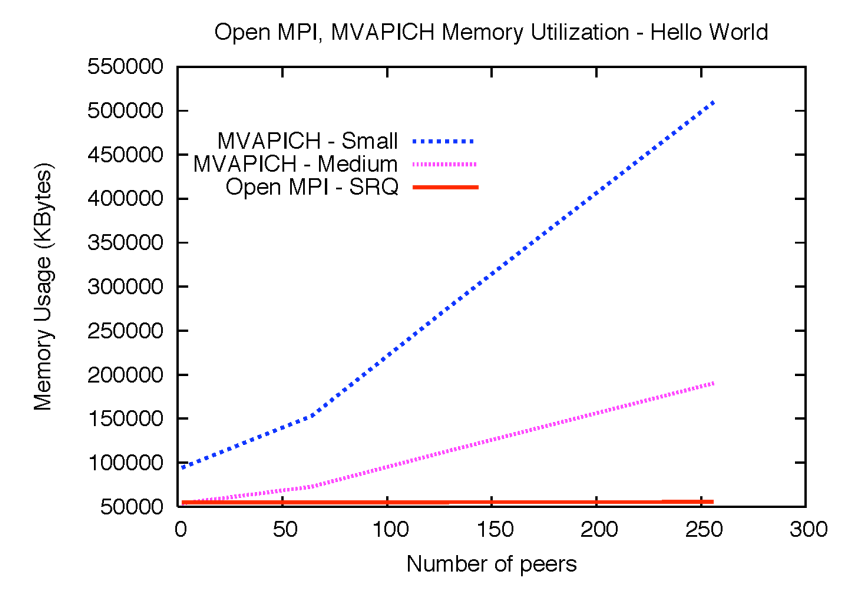

Fig. 1. Scaling of memory usage with number of processes in Open MPI and MVAPICH.

Open MPI is currently available on several InfiniBand Linux clusters at LANL including the institutional Coyote cluster, as well as the Roadrunner project phase 1 clusters Yellowrail and Redtail.

Open MPI and Roadrunner. As described elsewhere in this paper, the final Roadrunner system is an advanced Linux cluster of thousands of heterogeneous computational nodes, each comprising several AMD Opteron processors teamed with IBM Cell Broadband Engine accelerators. The network too, is heterogeneous, consisting of InfiniBand between computational nodes and a custom PCI Express-based link between Opteron and Cell BE "blades."

Although this is a highly complex and novel architecture, remarkably, Open MPI is sufficiently flexible and portable that prototype systems were quickly made available to application developers investigating algorithmic techniques and performance for Roadrunner. This proved crucial to the assessment phase of the Roadrunner project leading to the decision to go ahead with procurement of the final system.

Open MPI was also used as the prototype platform for the IBM software team developing new programming environments (DaCS and ALF) aimed at simplifying the task of application development for Roadrunner and future heterogeneous computer systems.

Work is currently in progress in collaboration with IBM to improve the scalability and performance of Open MPI for Roadrunner-scale systems. Also planned is improved support for the Roadrunner system architecture so that users can, if they choose, write a complete heterogeneous application using MPI rather than the new DaCS/ALF frameworks.

For more information contact David Daniel at ddd@lanl.gov.

[1] Message Passing Interface Forum, http://www.mpi-forum.org/.
[2] Open MPI, http://www.open-mpi.org/.
[3] The Los Alamos Message Passing Interface, http://public.lanl.gov/lampi/.
[4] G. Shipman et al., electronic *Proceedings 20th IEEE International Parallel and Distributed Processing Symposium,* (2006).

Funding Acknowledgments
- Department of Energy, National Nuclear Security Administration, Advanced Simulation and Computing Program